CLUBE

Maio/84

N.º 20

Coordenadores: MARIA IRENE • ALEXANDRE SOUSA • J. MAGALHÃES • AV. BOAVISTA, 832-2.º T. • TELEF. 65127 • 4100 PORTO

NESTE NÚMERO

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Ramos dos Santos & C.ª, Lda./Porto

Tiragem: 500 exemplares, Maio 1984

INTRODUÇÃO À LINGUAGEM MÁQUINA

ZX81/SPECTRUM

Autor: FERNANDO PRECES

(Cont. dos números anteriores)

PARTE II — SISTEMAS DE NUMERAÇÃO (Cont.)

2.4 — **Operações lógicas**

A nível dos números binários podemos efectuar operações lógicas, tais como:

AND, OR, XOR, NOT.

A operação AND (e) será verdadeira, somente quando os dois bits envolvidos estiveram em nível lógico (1).

EXEMPLO: A (AND) B

1.° Bit (A)	2.° Bit (B)	Operação AND
0	0	0
0	1	0
1	0	0
1	1	1

A operação OR (ou) será verdadeira, quando um dos bits, ou ambos, estiverem em nível lógico (1).

EXEMPLO: A (OR) B

1.° Bit (A)	2.° Bit (B)	Operação OR
0	0	0
0	1	1
1	0	1
1	1	1

A operação EXCLUSIVE ou (XOR) será verdadeira somente quando os dois bits envolvidos estiverem em nível lógico diferente.

EXEMPLO: A (XOR) B

1.° Bit (A)	2.° Bit (B)	Operação XOR
0	0	0
0	1	1
1	0	1
1	1	0

A operação NOT (Negação) inverte o nível lógico de um bit. Esta operação é representada pelo sinal ¯, colocado sobre o operando.

EXEMPLO $\overline{A}$

Bit (A)	Operação NOT
0	1
1	0

Vejamos um exemplo da aplicação destes sinais lógicos:

A = 1100 0101 (b) → 197 (d)
B = 0101 0111 (b) → 87 (d)

OPERAÇÕES LÓGICAS:

A (AND) B = 0100 0101 $_{(2)}$ → 69 $_{(10)}$
A (OR) B = 1101 0111 $_{(2)}$ → 215 $_{(10)}$
A (XOR) B = 1001 0010 $_{(2)}$ → 146 $_{(10)}$
(NOT A) = $\overline{A}$ = 0011 1010 $_{(2)}$ → 58 $_{(10)}$
(NOT B) = $\overline{B}$ = 1010 1000 $_{(2)}$ → 168 $_{(10)}$

2.5 — **Aritmética binária**

Para efectuarmos uma soma de dois bytes, adicionamos cada par de bits individualmente, até completarmos a soma dos seus 8 pares.
Essa soma é sempre realizada da direita para a esquerda, não esquecendo a existência de duas condições em que o **bit que sobra** será somado ao par de bits seguinte.

ADD (Soma)	VAI UM BIT
0 + 0 = 0	0
0 + 1 = 1	0
1 + 0 = 1	0
1 + 1 = 0	1
1 + 1 + 1 = 1	1

EXEMPLO DA SOMA:

```
  (1) (1)   (1) (1) (1)    ←  VAI UM BIT
      1100  0   1   0 1
  +   0101  0   1   1 1
  ---------------------
    1 0001  1   1   0 0
```

2.6 — **Binário aritmético absoluto**

Como já anteriormente foi referido, o processador Z80 manipula números binários que podem ir de 1 byte (8 dígitos binários — 8 bits), a 2 byts (16 dígitos binários — 16 bits).
Esses números são memorizados em binário absoluto, o que significa que cada byte está compreendido entre (0000 0000) e (1111 1111), que representam os números decimais entre (0 e 255).
Isto quer dizer, por outras palavras, que pelos processos normais não poderemos armazenar números que não sejam **números inteiros positivos.**

Experimente:

LET A = – 127 (número negativo)
LET B = 55.3896 (número com parte inteira e parte fraccionária)
POKE N, A
POKE (N + 1), B
PRINT N, PEEK N
PRINT (N + 1), PEEK (N + 1)

em que N é um endereço de opção, e obtém como resposta, (127 e 55).

No entanto, em linguagem máquina pode-se forçar o Z80 a trabalhar com uma diferente interpretação no valor de número. O método, chamado **2.° complemento aritmético,** é muito simples mas exige do processador uma vigilância muito apertada, graças a um programa de apoio que dirige o trabalho com esta forma aritmética.

Algumas instruções do Assembler Z80 trabalham o byte segundo esse conceito.

2.7 — **2.° complemento aritmético**

Um byte em 2.° complemento representa números entre (– 128 e + 127).

Para conseguir esta interpretação, o processador ignora o último bit (o oitavo bit a contar da direita), que passa a reconhecer apenas como um **bit de sinal.**

Se este tiver o valor "0", o número é considerado positivo; se o valor for "1", é considerado negativo.

Número Binário (2)	Decimal (10)	Hexa (16)
0 111 1111	+ 127	7F
0 111 1110	+ 126	7E
0 000 0010	+ 2	02
0 000 0001	+ 1	01
0 000 0000	0	00
1 111 1111	– 1	FF
1 111 1110	– 2	FE
1 000 0001	– 127	81
1 000 0000	– 128	80

QUADRO 2.6 — 1 byte em 2.° complemento

O processo para encontrar o 2.° complemento é muito simples. Vejamos um exemplo:

Encontrar o 2.° complemento de (– 45)

Método:

a) Formar o número binário, a partir do decimar absoluto.

45 (10) = 0010 1101 (2)

b) Encontrar o complemento deste número com uma operação lógica NOT.

$\overline{00101101}$ = 1101 0010

c) Formar o 2.° complemento, adicionando 1 ao resultado.

(1101 0010 + 0000 0001) =
1101 0011

Assim:

– 45 (10) = 11010011

O inverso do método descrito deve ser usado para converter números em 2.° complemento aritmético, no seu equivalente em binário absoluto.

PARTE III — COMO FUNCIONA O Z80

3.1 — **Introdução**

O microprocessador Z80, é um **chip** (nome que em inglês significa um componente de pequeno tamanho com uma imensidade de circuitos internos) que possui linhas de entrada (INPUT LINES) por onde chegam os impulsos e linhas de saída (OUTPUT LINES) que enviam os impulsos nele gerado para todos os auxiliares.

A sua estrutura interna pode ser dividida em 5 secções funcionais:

1 — Unidade de controlo
2 — Contador de programa
3 — Registro de instruções
4 — 24 registros para utilização
5 — Unidade aritmética e lógica

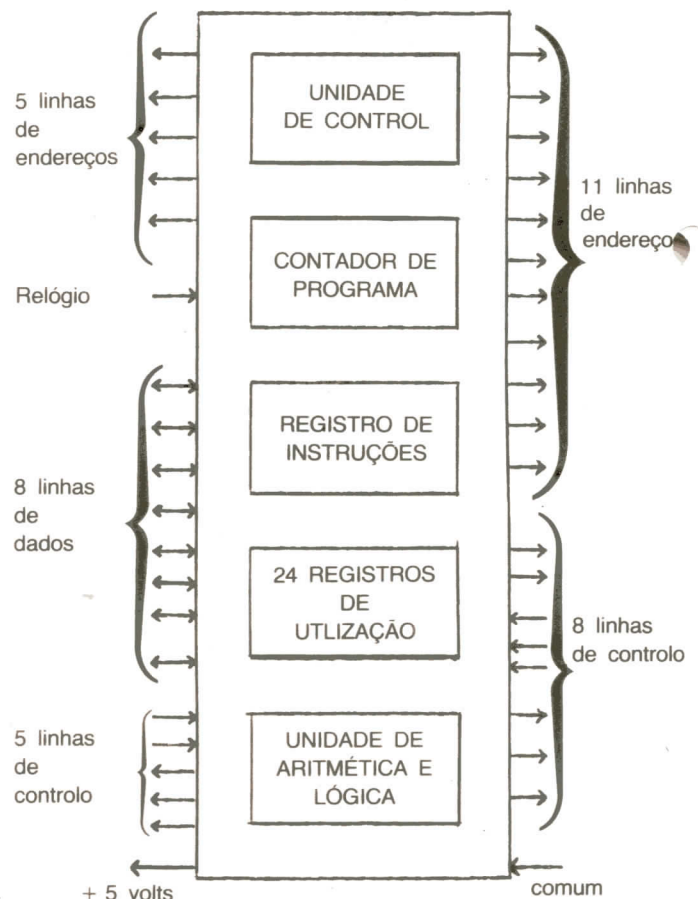



FIG. 3.1 — Uma versão simplificada da estrutura do Z80

3.2 — A unidade de controlo

A função deste bloco pode ser comparada, de uma forma simplista, ao encarregado geral de uma grande linha de montagem, cuja missão é controlar toda a sequência de execução; zelar pela existência da matéria-prima indispensável a cada operação (os dados), pela obtenção de um produto acabado de 1.ª qualidade e pelo envio deste para o destino certo.
No Z80 são gerados pelos vários circuitos um grande número de sinais, alguns para utilização interna, outros para o exterior através das linhas de controlo, com a missão de controlar com rigor toda a sequência do encaminhamento e execução de uma instrução.

POR EXEMPLO:

Se uma instrução analisada pelo Z80 pedir a leitura de um byte da memória, além de colocá-lo no ADDRESS BUS (trem de endereços), é colocado um sinal na linha de controlo chamada READ (leitura), para que a memória **saiba** que um byte deve ser lido e colocado no DATA BUS (trem de dados).

3.3 — Registro contador de Programa

O termo **registro,** que muitas vezes vai ser empregue, é usado para descrever um espaço-memória, interno do Z80, que pode albergar temporariamente o byte indicado.
Um registro duplo, tal como o nome sugere, é um duplo espaço-memória que poderá guardar qualquer endereço compreendido entre 0 e 65535 (representado em binário por 2 bytes — 16 bits).
Nesta conformidade o CONTADOR DE PROGRAMA é um registro duplo por dispor de 16 bits.
A sua missão é manter o controlo do endereço seleccionado, respeitante a uma instrução guardada em memória (quer da ROM ou da RAM), durante a execução do trabalho programado.
Sempre que uma instrução deva ser lida da memória, analisada e executada, o seu endereço é armazenado no registro P.C. (PROGRAM COUNTER).
Após a execução da instrução, o registro P.C. é incrementado (endereço n + 1) para apontar a instrução seguinte.
Quando o computador inicia a marcha após ser ligado a uma fonte de tensão, o registro P.C. arranca do endereço "0", sendo incrementado ao longo das instruções monitoras da ROM até finalizarem as **rotinas de iniciação.**
Depois disso, apesar do registro estar debaixo da alçada da unidade de controlo, existem instruções do Assembly para a sua manipulação **extra controlo.**

3.4 — Registro de instruções

É um espaço-memória de 8 bits, que tem como finalidade guardar uma cópia do código da instrução que vai ser executada. Neste registro também é a unidade de controlo que providencia a entrega da cópia que vem do endereço fixado pelo registro P.C.. Algumas instruções do Assembly podem alterar o processo.

3.5 — 24 registros para utilização.

24 espaços-memória, de 8 bits cada, foram reservados no Z80 para a comunicação directa deste com o exterior, através de uma adequada programação em código máquina.
Muitos dos nomes aplicados a estes registros que foram surgindo ao longo do processo evolutivo dos microprocessadores não são hoje nada significativos em relação às funções que desempenham. No entanto (esta é a opinião dos fabricantes), é preferível não os alterar, com vista a não dificultar os utilizadores.
Assim vamos encontrar por exemplo o registro "A", nome atribuído no passado, quando este **apenas** desempenhava a função de acumular resultados.
Apesar da maioria destes registros poderem trabalhar individualmente, executando funções bem definidas, sem interferência de uns para os outros, possuem também a faculdade de se poderem associar formando os chamados **registros duplos,** que se dividem em dois grupos:

a) Registros principais (8 pares)
b) » alternativos (4 pares)

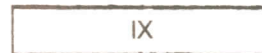



FIG. 3.5 — Os 24 registros de utilização

(Continua no próximo n.º)

FICHA DE CLÍNICA GERAL

O CLUBE Z80 tem recebido inúmeros pedidos para um trabalho colectivo sobre a criação da chamada FICHA DE CLÍNICA GERAL, tendo em vista a criação de um programa que use o *"Microdrive"* e que possa ser distribuído a todos os sócios que o solicitem.
Desta forma, agradecemos que todos os sócios interessados nos remetam um modelo do que consideram a **Ficha de Clínica Geral Ideal.**

CÁLCULO DE CUSTOS INDIRECTOS

In. BYTE, Fevereiro 1984

(Cont. do número anterior)

Continuando o artigo iniciado no n.º anterior, apresentamos agora o quadro 3 com os resultados finais da análise dos 4 departamentos do exemplo que focámos (o quadro contém todos os factores, as despesas iniciais, as quantias de cada factor e as quantias transferidas em e para cada departamento.

	A	B	C	D	E	F	G
1		DISTR. INDIRECTA DE CUSTOS METODO-CONTABILISTICO-					
2	------	------	------	------	------	------	------
3		14455	14108	15769	17861	62193	
4		DESPESAS TOTAIS NA LINHA ACIMA					
5	PARA//DE	--DPT A----	DPT B----	DPT C----	DPT D----	INICIAL--	TOTAL--
6	DPT A	0	1411	2365	2679	8000	14455
7	DPT B	2168	0	3154	1786	7000	14108
8	DPT C	0	1411	0	5358	9000	15769
9	DPT D	2168	2116	1577	0	12000	17861
10		DESPESAS LIQ. ABAIXO DESTA LINHA				36000	62193
11		10119	9170	8673	8038	36000	
12	------	------	------	------	------	------	------
13		FACTORES DE DISTRIBUICAO					
14		0	.1	.15	.15		
15		.15	0	.2	.1		
16		0	.1	0	.3		
17		.15	.15	.1	0		

QUADRO 3 — Apresentação dos resultados finais da análise dos 4 departamentos. Os totais em G6 até G10 correspondem respectivamente aos de B3 até F3.

Agora podem ser analisadas as despesas totais de cada departamento, concluindo o que fazer para as reduzir.

A análise de balanços, o mapa de orçamentos ou o mapa de encomendas podem ser feitos e alterados com extrema simplicidade.

MODIFICAR OS DADOS

Após rever os resultados de cada conjunto de cálculos, pode acontecer querer alterar um ou mais factores. Isso é possível através do uso posterior da tecla de Recalcular para obter novos resultados. Neste caso, e contrariamente ao que se afirmou antes, alguns valores diminuirão. Todavia, as alterações terão que seguir uma linha determinada e não oscilar para cima e para baixo. E, como os cálculos se fazem facilmente, pode voltar ao princípio estabelecendo nova tabela e completando novamente o processo.

NOTA: Não será redundante chamar a atenção para os programas do tipo VI-SICALC, cuja mais conhecida aplicacão é a obtenção de resultados (simulados ou não), a partir de dados parciais.

USAR O BASIC

A listagem que se apresenta a seguir contém o programa em BASIC para o programa exemplificado.

```
  10 REM programa p/distribuir c
ustos
  15 REM adaptado p/Alexandre So
usa
  20 LET n=4
  30 DATA 0,.1,.15,.15
  40 DATA .15,0,.2,.1
  50 DATA 0,.1,0,.3
  60 DATA .15,.15,.1,0
  70 DATA 8000,7000,9000,12000
  72 DATA 0,1411,2365,2679
  74 DATA 2168,0,3154,1786
  76 DATA 0,1411,0,5358
  78 DATA 1544,1543,982,0
  80 DIM I(4)
  82 DIM N(4)
  84 FOR K=1 TO 4
  85 LET N(K)=0
  86 NEXT K
  90 DIM M(4)
 100 DIM T(4)
 110 DIM P(4,4)
 130 FOR R=1 TO N
 140 FOR C=1 TO N
 150 READ V
 160 LET P(R,C)=V
 170 NEXT C
 180 NEXT R
 200 FOR Z=1 TO N
 210 READ I(Z)
 220 NEXT Z
 240 FOR Z=1 TO N
 250 LET M(Z)=I(Z)
 260 NEXT Z
 280 FOR Z=1 TO N
 290 LET T(Z)=I(Z)
 300 NEXT Z
 320 FOR R=1 TO N
 330 LET G=0
 340 FOR C=1 TO N
 350 LET D=M(C)*P(R,C)
 360 LET G=G+D
 370 NEXT C
 380 LET M(R)=G+I(R)
 390 PRINT R,M(R)
 400 NEXT R
 420 LET E=0
 430 LET F=0
 440 FOR Z=1 TO N
 450 LET E=E+M(Z)
 460 LET F=F+T(Z)
 470 NEXT Z
 480 PRINT E,F,E-F
 500 FOR Z=1 TO N
 510 LET T(Z)=M(Z)
 520 NEXT Z
 530 REM TESTE PARA COMPLETAR CI
CLOS ITERATIVOS
 540 IF (E-F)>=1 THEN GO TO 320
 550 FOR C=1 TO N
 560 LET N(C)=T(C)
 570 FOR R=1 TO N
 580 LET N(C)=N(C)-T(C)*P(R,C)
 590 NEXT R
 600 NEXT C
 610 FOR Z=1 TO N
 620 PRINT INT (I(Z)),INT (T(Z))
```

```
,INT (N(Z))
630 NEXT Z
640 PRINT "TERMINADO"
650 REM IMPRIMIR RESULTADOS
660 PRINT "TOTAL DE DESPESAS="
670 FOR Z=1 TO N
680 PRINT "dept ";z;" =";INT (T
(Z)),
690 NEXT Z
700 PRINT " "
710 PRINT
720 PRINT "VAL.DISTRIB.-TOTAL="
730 FOR R=1 TO N
740 FOR C=1 TO N
750 PRINT INT (P(R,C)*T(C)),
760 NEXT C
770 PRINT INT (I(R)),INT (T(R))
780 NEXT R
790 PRINT "DESP.LIQUID.="
800 LET Q=0
810 FOR C=1 TO N
820 LET Q=Q+N(C)
830 PRINT "dept ";c;" =";INT (N
(C)),
840 NEXT C
850 PRINT INT (Q)
```

Todos os valores intermédios são escritos no écran e as respostas finais terão o aspecto apresentado no quadro 3.
Para escrever os totais intermédios, pode eliminar as instruções PRINT nas linhas 390 e 480. Pode também fazer anotações para melhor compreensão ou incluir várias instruções numa linha — mas ficará muito mais complicado.
Se quiser incluir mais departamentos, terá que acrescentar mais instruções de dados, e as instruções de dimensionamento (linhas 10-110) terão de ser alargadas até cobrirem o número máximo de departamentos.
As instruções PRINT deverão também ser alteradas para acompanhar esse aumento. Se, por exemplo, houver 100 departamentos, terão que ser reservados 10 000 espaços para os factores e 10 000 para as quantias distribuídas. A uma razão de 2 bytes por espaço, esta análise exige pelo menos 40 000 bytes de armazenamento. Mais uma vez poderíamos fazer reduções significativas, limitando o armazenamento apenas aos factores realmente necessários e desprezando a simetria das tabelas. No entanto, repare-se que os valores elevados exigiram cálculos de dupla precisão e um aumento correspondente em instruções de armazenamento.

PROCEDIMENTO EM MÁQUINAS DE CALCULAR

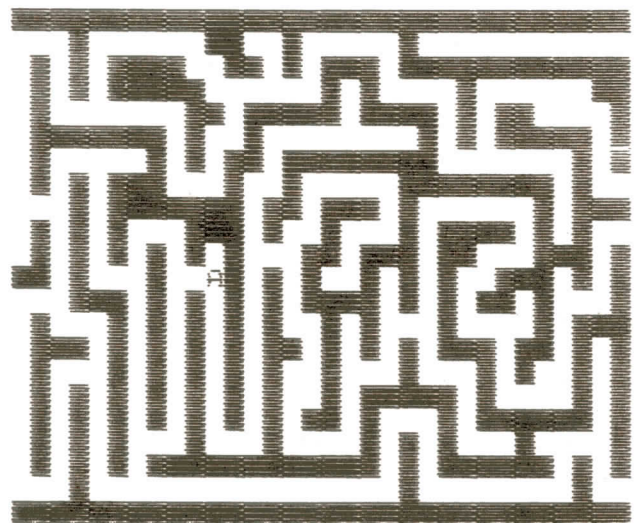
Com uma calculadora e papel, pode simular o procedimento do computador, começando com a tabela 1.
Use um lápis macio para escrever as quantidades de distribuição e o total do departamento (coluna G). Depois substitua-os quando conseguir melhores cálculos. Os totais são também copiados para o topo de cada coluna (linha 3) para ser mais fácil encontrar o par de números incluídos em cada cálculo dos valores de distribuição.

CAPACIDADE DO COMPUTADOR

A resolução de um problema através deste método depende do número máximo de colunas que as folhas electrónicas de cálculo contém e/ou da capacidade total do computador. Por exemplo, se usar um programa de folhas electrónicas de cálculo ou um programa que armazene a tabela completa de valores distribuídos e a de factores, precisará de muito mais capacidade do que se usar apenas os elementos não-zero.

Dado que, nos casos reais, cerca de 75 % dos factores equivalem a zero, pode re-arranjar e condensar os dados se não quiser ter todos os itens ordenados em linhas e colunas por departamento.
Por exemplo, num caso de 91 departamentos, só existiam 620 factores dos 8291 (91×91) possíveis. Assim, apenas tiveram lugar 7,5 % das transferências possíveis. Este exemplo foi analisado usando um DOS 2.0 e LOTUS 1-2-3 num computador IBM (PC) com 327 680 bytes. Inicialmente, havia 211 340 bytes livres. Contudo, os dados dos 91 departamentos não caberiam neste espaço; seriam necessários 64 K bytes adicionais ou um total de 280 K bytes do espaço para o exemplo.
Compare estes dados com os que seriam necessários, para o mesmo caso em BASIC, com códigos de factores não-zero. Aí bastariam 11 500 bytes para os dados, programa, despesas iniciais, finais e líquidas. Mesmo com o armazenamento da tabela total, o problema poderia ser resolvido com os 61 K bytes de espaço livre num IBM (PC). Bastariam 8 iteracções para obter a solução: cerca de $20 milhões em despesas totais. Este valor está aproximado até ao último dólar.
Realmente é possível um grande rigor numa razão de apenas uma iteracção por coluna e apenas a codificação para factores não-zero é suficiente para executar o programa.
Por último, chamamos a atenção dos contabilistas que desconhecem o uso dos microcomputadores em cálculos comerciais. Aqui está uma boa prova da sua utilização.

SPECTRUM MAZE SPECTRUM 16 K

In. ZX COMPUTING, Fevereiro/Março 83
Adapt. e Trad.: J. MAGALHÃES

Trata-se de um labirinto com movimentos para a direita ou para a esquerda (scroll) com o fim de fazer percorrer a letra "A" por todo o labirinto até encontrar a única saída.

CONSIDERAÇÕES SOBRE O PROGRAMA

Será bastante simples alterar uma parte do programa. Cada «DATA statment» de 2000 a 2117 controla uma coluna do écran.

Estando o écran dividido em 3 grupos de 8 linhas, cada quadrado da coluna está codificado num dígito binário: o quadrado preto é 1 e o quadrado branco é 0 (zero). Finalmente cada grupo de 8 quadrados está convertido num número decimal entre 0 (zero) e 255. Repare na fig.:

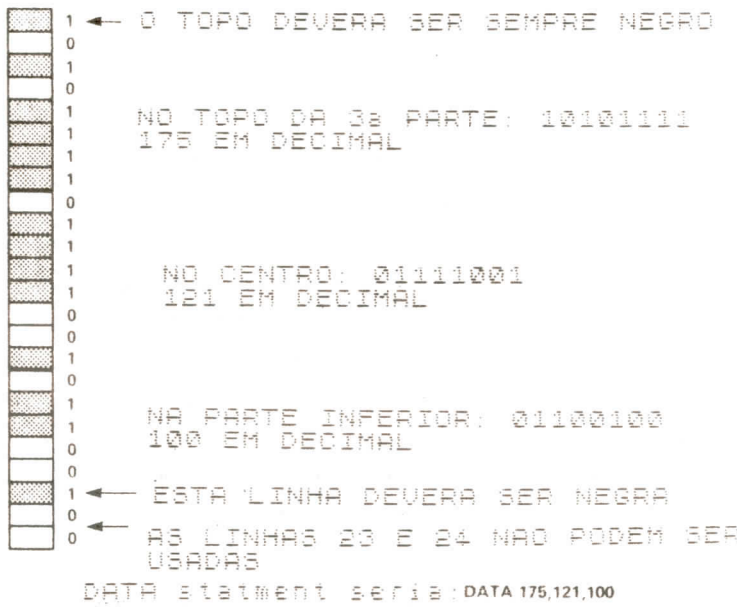



Fig. 1. Uma coluna do ecran

Quatro das rotinas em código máquina podem ser usadas nos seus programas:

Linha 1 — Move as cores um caracter à esquerda. 23760 é o 1.º endereço.

Linha 2 — Move as cores um caracter à direita. 23798 é o 1.º endereço.

Linha 3 — Move lentamente estes caracteres para a esquerda em um caracter ("scroll" um "pixel" de cada vez em 8 vezes). 23836 é o 1.º endereço.

Linha 4 — Move os caracteres da mesma forma que na linha 3, mas para a direita. 23876 é o 1.º endereço.

A 5.ª linha é também em código máquina mas não pode ser aplicada fora desde programa.

Se pretender as rotinas acima indicadas, dê entrada das linhas 1 a 4, 6 a 9 e 100 a 460 e faça RUN. Apague todas as linhas da 6 em diante (o c.m. ficará no lugar da REM).

COMO JOGAR

Para movimentar o caracter "A" use as teclas 5,6,7 e 8. Os possuidores do SPECTRUM 48K podem continuar o labirinto até 32 écrans de comprimento, o que significa serem necessárias mais de 1000 "DATA statments".

NOTA: O caracter "A" pode ser trocado por qualquer outro caracter.

```
   1 REM zzzzzzzzzzzzzzzzzzzzzzzzz
zzzzzzzzz
   2 REM zzzzzzzzzzzzzzzzzzzzzzzzz
zzzzzzzzz
   3 REM zzzzzzzzzzzzzzzzzzzzzzzzz
zzzzzzzzzzz
   4 REM zzzzzzzzzzzzzzzzzzzzzzzzz
zzzzzzzzz
   5 REM zzzzzzzzzzzzzzzzzzzzzzzzz
zzzzzzzzz
   6 RESTORE 6: FOR g=0 TO 28: R
EAD h: POKE 23760+g,h: NEXT g
   7 FOR g=0 TO 28: READ h: POKE
 23798+g,h: NEXT g
   8 FOR g=0 TO 33: READ h: POKE
 23836+g,h: NEXT g
   9 FOR g=0 TO 25: READ h: POKE
 23876+g,h: NEXT g
  10 FOR G=0 TO 9: READ A: POKE
23914+G,A: NEXT G
  11 RESTORE 2000: FOR H=0 TO 31
: FOR g=0 TO 21
  12 IF g=0 OR g=8 OR g=16 THEN
READ a: POKE 23915,a
  13 LET x=g-(g>=8)*8-(g>=16)*8
  14 PRINT AT g,h;: POKE 23917,1
27-x*8
  15 RANDOMIZE USR 23914
  16 NEXT g: NEXT h
  20 GO TO 500
 100 DATA 13,17,0,88,33,1,88,1,2
55,2
 110 DATA 237,176,33,31,88,14,24
 120 DATA 58,141,92,17,32,0,119,
25
 130 DATA 13,200,24,250
 200 DATA 13,17,255,90,33,254,90
,1,255,2
 210 DATA 237,184,33,0,88,14,24
 220 DATA 58,141,92,17,32,0,119,
25
 230 DATA 13,200,24,250
 300 DATA 6,8,33,31,64,22,0,30
 310 DATA 65,14,32,175,126,23
 320 DATA 119,43,13,32,249,25,62
 330 DATA 88,188,40,3,43,24,235
 340 DATA 5,175,184,32,225,201
 400 DATA 6,8,33,0,64,14,32,175
 410 DATA 126,31,119,35,13,32,24
9
 420 DATA 62,88,188,32,241,5,175
 430 DATA 184,32,233,201
 450 DATA 62,0,203,127,40,3,62
 460 DATA 143,215,201
 500 LET y=0: : LET x=11
 510 POKE 23672,0: POKE 23673,0:
 POKE 23674,0
 520 DEF FN a()=INT ((PEEK 23672
+PEEK 23673*256+PEEK 23674*65535
)/50)
 580 IF y=107 THEN PRINT AT 10,
5;"PARABÉNS!";AT 12,16;"COMPLETO
U";AT 14,17;"O LABIRINTO EM";AT
19,18;
 590 IF y=107 THEN PRINT INT ((P
EEK 23672+PEEK 23673*256+PEEK 23
674*65535)/50);" SEGUNDOS.": STO
P
 600 PRINT AT x,10;"A"
 610 LET a$=INKEY$
 620 IF a$="5" AND POINT (74,175
-8*x)<>1 THEN GO SUB 8500
 630 IF a$="8" AND POINT (90,175
-8*x)<>1 THEN GO SUB 8000
 640 IF a$="7" AND POINT (82,183
-8*x)<>1 THEN PRINT AT x,10;" ":
 LET x=x-1: GO TO 600
 650 IF a$="6" AND POINT (82,167
-8*x)<>1 THEN PRINT AT x,10;" ":
 LET x=x+1: GO TO 600
 660 GO TO 580
1999 DATA 0,0,0
2000 DATA 255,255,255
2001 DATA 128,16,4
2002 DATA 191,119,244
2003 DATA 132,2,4
2004 DATA 245,250,252
2005 DATA 132,8,4
2006 DATA 181,239,247
2007 DATA 181,128,4
2008 DATA 183,191,215
```

```
2009 DATA 176,128,20
2010 DATA 158,239,244
2011 DATA 200,192,20
2012 DATA 227,255,212
2013 DATA 174,0,20
2014 DATA 136,223,244
2015 DATA 235,130,20
2016 DATA 138,56,84
2017 DATA 186,239,212
2018 DATA 162,136,20
2019 DATA 186,190,244
2020 DATA 138,32,132
2021 DATA 235,251,188
2022 DATA 175,0,132
2023 DATA 161,126,244
2024 DATA 189,98,20
2025 DATA 161,75,212
2026 DATA 173,88,84
2027 DATA 173,19,116
2028 DATA 165,254,84
2029 DATA 164,32,68
2030 DATA 167,247,220
2031 DATA 176,132,4
2032 DATA 190,191,244
2033 DATA 162,130,36
2034 DATA 184,184,180
2035 DATA 141,143,148
2036 DATA 229,32,212
2037 DATA 133,52,20
2038 DATA 189,227,212
2039 DATA 160,136,20
2040 DATA 190,175,244
2041 DATA 128,130,4
2042 DATA 238,250,244
2043 DATA 138,16,4
2044 DATA 187,215,252
2045 DATA 160,80,4
2046 DATA 173,87,244
2047 DATA 165,16,132
2048 DATA 181,118,188
2049 DATA 161,54,188
2050 DATA 140,22,188
2051 DATA 231,246,188
2052 DATA 178,38,132
2053 DATA 178,160,180
2054 DATA 134,190,180
2055 DATA 180,130,180
2056 DATA 145,190,180
2057 DATA 223,0,52
2058 DATA 136,127,164
2059 DATA 187,0,172
2060 DATA 169,255,172
2061 DATA 172,192,44
2062 DATA 166,94,236
2063 DATA 176,194,140
2064 DATA 163,154,188
2065 DATA 186,186,132
2066 DATA 162,10,244
2067 DATA 170,250,4
2068 DATA 170,67,244
2069 DATA 139,26,4
2070 DATA 187,250,252
2071 DATA 162,2,4
2072 DATA 174,187,180
2073 DATA 160,162,36
2074 DATA 191,175,116
2075 DATA 188,40,68
2076 DATA 161,227,220
2077 DATA 175,8,68
2078 DATA 162,123,244
2079 DATA 184,216,132
2080 DATA 142,139,188
2081 DATA 162,169,132
2082 DATA 250,43,52
2083 DATA 139,234,116
2084 DATA 160,72,196
2085 DATA 191,94,92
2086 DATA 160,67,76
2087 DATA 175,121,100
2088 DATA 161,109,52
2089 DATA 189,73,148
2090 DATA 161,91,212
2091 DATA 175,66,20
2092 DATA 165,91,116
2093 DATA 177,73,68
2094 DATA 189,109,92
2095 DATA 169,9,68
2096 DATA 171,227,84
2097 DATA 160,63,84
2098 DATA 190,224,84
2099 DATA 168,175,244
2100 DATA 171,162,36
2101 DATA 168,58,172
2102 DATA 175,162,36
2103 DATA 160,174,180
2104 DATA 190,160,164
2105 DATA 130,191,180
2106 DATA 186,160,4
2107 DATA 186,183,188
2108 DATA 162,160,4
2109 DATA 190,191,116
2110 DATA 128,132,68
2111 DATA 255,245,220
2112 DATA 128,4,68
2113 DATA 191,223,212
2114 DATA 164,80,20
2115 DATA 173,123,244
2116 DATA 160,0,4
2117 DATA 191,255,252
7999 DATA 0,0,0
8000 PRINT AT X,10;" "
8010 RANDOMIZE USR 23760: RANDOM
IZE USR 23836
8020 PRINT AT X,10;"A"
8030 LET Y=Y+1: RESTORE 2031+Y:
FOR G=0 TO 21: IF G=0 OR G=8 OR
G=16 THEN READ A: POKE 23915,A
8032 LET J=G-(G>=8)*8-(G>=16)*8
8034 POKE 23917,127-J*8
8035 PRINT AT G,31;
8036 RANDOMIZE USR 23914: NEXT G
: RETURN
8500 PRINT AT X,10;" "
8510 RANDOMIZE USR 23798: RANDOM
IZE USR 23876
8520 PRINT AT X,10;"A"
8530 LET Y=Y-1: RESTORE 2000+Y:
FOR G=0 TO 21: IF G=0 OR G=8 OR
G=16 THEN READ A: POKE 23915,A
8532 LET J=G-(G>=8)*8-(G>=16)*8
8534 POKE 23917,127-J*8
8535 PRINT AT G,0;
8536 RANDOMIZE USR 23914: NEXT G
: RETURN
```

ROTINA EM CÓDIGO MÁQUINA — Quem responde?

```
LDE,N
LDD,N
LDA,("at"(22))N
RST16
LDA,D
RST16
LDA,E
RST16
LDA,"INK"(16)N
RST16
LDA,N  (car)
RST16
LDA,N  (simbolo)
RST16
RET
```

"A rotina que listo ao lado, como podem ver, coloca em certa posição do écran um determinado símbolo com a côr que se pretende. O problema é que, após o símbolo, surge sempre um número que varia de linha para linha. Se souberem, gostaria que me explicassem porque é que isso acontece e como é possível eliminar tal número do écran".

MÁRIO REBELO/COIMBRA

Escrevam ao CLUBE Z80 respondendo à pergunta do Mário Rebelo. Publicaremos as vossas explicações em Junho.

GRAFIT - Gráficos xx - yy

16 e 48 K

Autor: PAUL BATES
Adapt. por: J. MAGALHÃES e ALEXANDRE SOUSA

Este programa permite o traçado gráfico a partir de informação de vários estilos e pode ser usado com mais de uma forma sem obrigar a reentrar os dados. Possui mais do que um menu para ajudar o utilizador.

O programa ocupa cerca de 5K e está totalmente em Basic.

Se possuir uma printer Sinclair ou Timex pode copiar o gráfico para o papel.

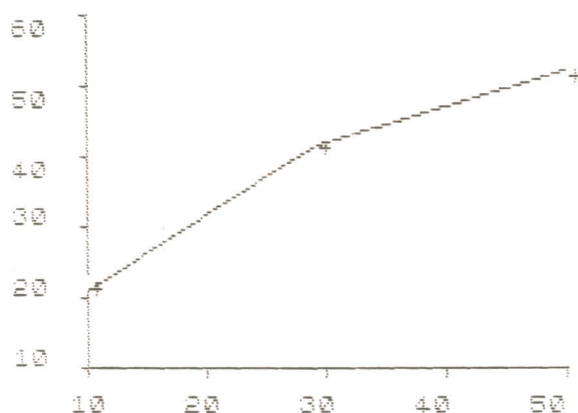
Possui uma opção que dá a equação da regressão linear, e desenha a recta que melhor se aproxima das características dos dados obtidos. O coeficiente de regressão indica a melhor ou pior aproximação dos dados em termos da confiança estatística.

r = 0 — indica que os dados não seguem uma relação linear.
r = 1 — indica que os dados traduzem na perfeição a equação determinada.
r = .7 — indica que os dados estão dentro de uma boa margem de aceitação.

SOBRE O PROGRAMA

LINHAS		
	20 — 140	PREPARAR O MENU
	1000 — 1600	ENTRADA DOS DADOS
	2000 — 2140	OPÇÃO DOS SÍMBOLOS
	3000 — 3080	TRAÇADO DO GRÁFICO
	4000 — 4040	DESENHO DOS EIXOS
	4040 — 4320	DESENHO DAS MARCAS
	4330 — 4420	DESENHO DOS NOMES DOS EIXOS
	4420 — 4520	DESENHO DOS PONTOS
	4700 — 4760	DESENHO DA LIGAÇÃO DOS PONTOS
	4800 — 4830	PEDIDO DA CÓPIA
	5000 — 5300	REGRESSÃO LINEAR

```
TITULO:

  X        Y
 ---      ---
 10       20
 30       40
 50       50
```

```
     REGRESSAO LINEAR

 NOME DO TESTE E:

 O DECLIVE E: 0.75

 A INTERCEPCAO E: 14.166667

 DAI A EQUACAO >

    Y=0.75* X  + 14.166667

 O COEFICIENTE DE CORRELACAO E:
0.9819805

 COPIA S/N ?
```

```
        MENU

 1-ENTRADA DE DADOS

 2-ESCOLHER DESENHO DE GRAFICO

 3-ESCOLHER TIPO DE TRACO

 4-DESENHO DE GRAFICO

 5-REGRESSAO

 6-FIM do Programa

 Optar p/num. ? -0 p/ regr.MENU
```

```
  10   BORDER 3: PAPER 6: INK 1:
 CLS
  11 POKE 23658,8
  30 PRINT AT 3,11;"MENU"
  40 PRINT AT 5,2;"1-ENTRADA DE
DADOS"
  50 PRINT AT 7,2;"2-ESCOLHER DE
SENHO DE GRAFICO"
  60 PRINT AT 9,2;"3-ESCOLHER TI
PO DE TRACO"
  70 PRINT AT 11,2;"4-DESENHO DE
 GRAFICO"
  80 PRINT AT 13,2;"5-REGRESSAO"
  90 PRINT AT 15,2;"6-FIM do Pro
grama"
 100 PRINT AT 18,1;"Optar p/num.
 ? -0 p/ regr.MENU"
 110 INPUT Z$
 120 IF Z$>"6" THEN GO TO 110
 125 IF Z$="6" THEN STOP
 130 LET A=VAL Z$
 140 CLS
 150 IF A=0 THEN GO TO 10
 160 GO TO A*1000
1000 BORDER 2: PAPER 1: INK 7: C
LS
1010 DIM B(50): DIM X(50): DIM Y
(50): DIM A$(10): DIM B$(10): DI
M C$(10): DIM D$(1)
1020 PRINT AT 1,5;"ENTRADA DE DA
DOS"
1025 LET G=0: LET H=0
1030 PRINT AT 3,2;"N. DE PARES :
"
1040 BEEP .5,20: INPUT B
1045 IF B=0 THEN GO TO 1500
1050 PRINT AT 3,22;B
```

```
1060 FOR I=1 TO 8
1070 PRINT AT 5,5;"X";I;"="
1080 INPUT X(I)
1090 PRINT AT 5,9;X(I)
1100 BEEP .1,20
1110 PRINT AT 10,5;"Y";I;"="
1120 INPUT Y(I)
1130 PRINT AT 10,9;Y(I)
1140 BEEP .1,20
1150 PRINT AT 5,9;"    ": PRINT
AT 10,9;"    "
1160 NEXT I
1170 PRINT AT 12,1;"NOME DO GRAF
ICO:"
1180 INPUT A$
1190 PRINT AT 12,14;A$
1200 PRINT AT 14,1;"EIXO-X: "
1210 INPUT B$
1220 PRINT AT 14,15;B$
1230 PRINT AT 16,1;"EIXO-Y: "
1240 INPUT C$
1250 PRINT AT 16,15;C$
1260 PRINT AT 18,1;"MIN-X= "
1270 INPUT C
1280 PRINT AT 18,7;C
1290 PRINT AT 18,12;"MAX-X= "
1300 INPUT D
1310 PRINT AT 18,18;D
1320 PRINT AT 20,1;"MIN-Y= "
1330 INPUT E
1340 PRINT AT 20,7;E
1350 PRINT AT 20,12;"MAX-Y= "
1360 INPUT F
1370 PRINT AT 20,18;F
1380 PAUSE 100
1390 CLS
1400 PRINT AT 5,2;"INTERVALO X:
"
1410 INPUT Y
1420 PRINT AT 5,25;Y
1430 PRINT AT 10,2;"INTERVALO Y:
"
1440 INPUT Z
1450 PRINT AT 10,25;Z
1460 PAUSE 100
1500>CLS : PRINT AT 5,1;"DESEJA
A COPIA DOS"
1510 PRINT AT 7,1;"PARES DE DADO
S?"
1520 PRINT AT 11,2; INVERSE 1; F
LASH 1;"(S)IM OU (N)AO"
1530 INPUT D$
1540 IF D$<>"S" THEN GO TO 10
1542 LPRINT TAB 6;"TITULO: ";A$
1544 LPRINT
1550 LPRINT TAB 10;"X";"
  ";"Y"
1560 LPRINT TAB 9;"---";"
  ";"---"
1570 FOR I=1 TO 8
1580 LPRINT TAB 9;X(I);"
  ";Y(I)
1590 NEXT I
1600 GO TO 10
2000 BORDER 4: PAPER 6: INK 2: C
LS
2020 PRINT AT 1,5;"CARACTERES":
LET G=0
2030 PRINT AT 4,2;"OPCAO 1: ."
2040 PRINT AT 8,2;"OPCAO 2: *"
2050 PRINT AT 12,2;"OPCAO 3: +"
2060 PRINT AT 14,1; INVERSE 1; F
LASH 1;"QUAL A SUA OPCAO? "
2070 INPUT G
2080 IF G=1 THEN LET G=46
2090 IF G=2 THEN LET G=42
2100 IF G=3 THEN LET G=43
2110 PRINT AT 18,2;"A SUA OPCAO
E "
2120 PRINT AT 18,16;CHR$ G
2130 PAUSE 150
2140 GO TO 10
3000 BORDER 5: PAPER 7: INK 2: C
LS
3010 PRINT AT 1,5;"OPCAO PLOT":
LET H=0
3020 PRINT AT 5,2;"OPCAO 1: APEN
AS OS PONTOS"
3030 PRINT AT 10,2;"OPCAO 2: PON
TOS UNIDOS POR"
3040 PRINT AT 11,12;"LINHAS RECT
AS"
3050 PRINT AT 15,4; INVERSE 1; F
LASH 1;" A SUA OPCAO?"
3060 INPUT H
3070 IF H<1 OR H>2 THEN GO TO 30
50
3080 GO TO 10
4000 IF G=0 THEN LET G=46
4002 IF H=0 THEN LET H=1
4005 BORDER 5: PAPER 7: INK 0: C
LS
4010 LET I=48: LET J=32
4020 PLOT I,J: DRAW 0,120
4030 PLOT I,J: DRAW 200,0
4040 LET K=17
4050 PLOT I,J: DRAW -3,0
4060 LET L=(F-E)/Z
4070 FOR A=1 TO L
4080 LET J=J+INT (120/L)
4090 PLOT I,J: DRAW -3,0
4100 NEXT A
4110 IF F>0 THEN LET L=(F-E)/Z:
IF F<0 THEN LET L=(((E-F)/Z)*-1)
4115 LET Q=E
4120 FOR I=1 TO L+1
4130 PRINT AT K,2;Q
4140 LET Q=(Q+Z)
4150 LET K=(K-(17/(L+1)))
4160 NEXT I
4170 LET I=48: LET J=32
4180 LET L2=(D-C)/Y
4190 PLOT I,J: DRAW 0,-3
4200 FOR A=1 TO L2
4210 LET I=I+INT (200/L2)
4220 PLOT I,J: DRAW 0,-3
4230 NEXT A
4240 IF D>0 THEN LET L2=(D-C)/Y:
 IF D<0 THEN LET L2=((D-C)/Y)*-1
4250 LET K2=5: LET W=C
4260 FOR A=1 TO L2+1
4270 PRINT AT 19,K2;W
4280 LET W=(W+Y)
4290 LET K2=INT K2+(30/(L2+1))
4300 NEXT A
4310 PRINT AT 21,8;B$
4320 PRINT AT 0,8;A$
4330 REM TITULO DO EIXO DO EIXO
DE Y"
4340 LET M=6
4350 FOR I=1 TO 10
4360 PRINT AT M,0;C$(I)
4370 LET M=M+1
4380 NEXT I
4390 LET N=0: LET N=N+C
4400 LET N2=0: LET N2=N2+E
4420 REM PLOT DOS PONTOS
4430 LET P0=32: LET O0=48
4440 FOR A=1 TO 8
4450 LET O=((X(A)-N)/(D-N))
4460 LET O=INT ((O*25)+6)
4470 LET P=((Y(A)-N2)/(F-N2))*15
4480 LET P=INT ((P-17)*-1)
4490 PRINT AT P,O;CHR$ G
4500 IF H=2 THEN GO TO 4700
4510 NEXT A
4520 GO TO 4800
4700 LET Z5=1
4710 LET P1=INT (((P/21)-1)*-175
): LET O1=(O*8)
4720 PLOT O0,P0
4730 DRAW (O1-O0),(P1-P0)
```

```
4740 LET P0=P1: LET O0=O1
4750 LET Z5=Z5+1: IF Z5=B THEN G
O TO 4800
4760 GO TO 4510
4800 DIM E$(1): INPUT "COPIA(S/N
)?";E$
4810 IF E$="N" THEN GO TO 10
4820 IF E$="S" THEN COPY
4830 GO TO 10
5000 BORDER 6: PAPER 4: INK 0: C
LS
5010 PRINT AT 1,6;"[illegible]
[illegible]"
5020 PRINT AT 4,2;"NOME DO TESTE
 E: ";A$
5030 LET D3=0: LET E3=0: LET F3=
0: LET G3=0: LET H3=0
5040 FOR I=1 TO B
5070 LET D3=D3+X(I)
5080 LET E3=E3+Y(I)
5090 LET F3=F3+(X(I)*X(I))
5100 LET G3=G3+(Y(I)*Y(I))
5110 LET H3=H3+(X(I)*Y(I))
5120 NEXT I
5130 LET L3=H3-((D3*E3)/B)
5140 LET M3=F3-((D3*D3)/B)
5150 LET N3=L3/M3
5160 LET O3=D3/B
5170 LET P3=E3/B
5180 LET Q3=SQR (M3*(G3-((E3*E3)
/B)))
5190 LET R3=L3/Q3
5200 LET S3=P3-(N3*O3)
5210 PRINT AT 6,2;"O DECLIVE E:
";N3
5220 PRINT AT 8,2;"A INTERCEPCAO
 E: ";S3
5230 PRINT AT 10,2;"DAI A EQUACA
O > "
5240 PRINT AT 12,5;"Y=";N3;"* X
 + ";S3
5250 PRINT AT 14,1;"O COEFICIENT
E DE CORRELACAO E: ";R3
5260 PRINT AT 18,2;"[illegible]
"
5270 DIM J$(1): INPUT J$
5280 IF J$="S" THEN COPY
5290 IF J$<>"S" THEN GO TO 10
5300 GO TO 10
```

CARACTERES

OPCAO 1: .

OPCAO 2: *

OPCAO 3: +

QUAL A SUA OPCAO?

OPCAO PLOT

OPCAO 1: APENAS OS PONTOS

OPCAO 2: PONTOS UNIDOS POR
LINHAS RECTAS

A SUA OPCAO?

X	Y
120	10
125	20
130	38
135	25
140	7

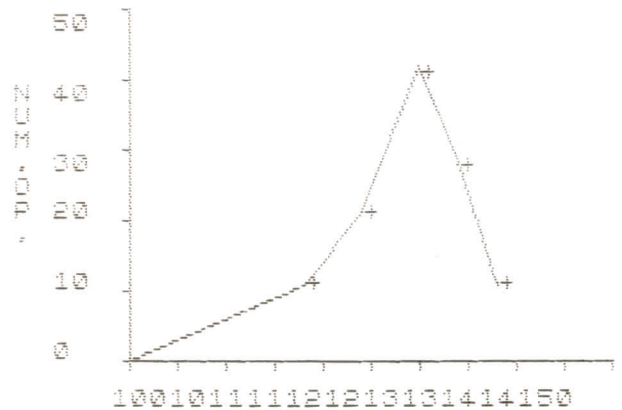
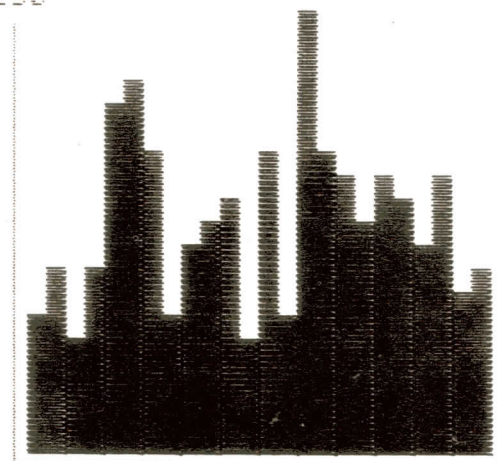



GRAFICO DE BARRAS

12 MESES -COMPRAS/VENDAS

QUEM USAR TV PRETO/BRANCO

MODIFICAR LINHA 123
para INK 4 e retirar INK e PAPER
das linhas 71 e 140

(Autor:
Alexandre Sousa)

```
10 DIM C(12): DIM V(12)
40 FOR g=1 TO 12
50 INPUT "VENDAS ";V(g)
60 INPUT "COMPRAS ";C(g)
62 IF G=1 THEN LET YMAX=V(1):
GO TO 70
64 IF V(G)>YMAX THEN LET YMAX=
V(G)
66 IF C(G)>YMAX THEN LET YMAX=
C(G)
```

```
  70 NEXT g
  71 CLS : INK 7: PAPER 7
  72 GO SUB 500
  74 LET K=10: LET J=12: PLOT K,
J: DRAW 0,150: PLOT K,J: DRAW 20
0,0
  76 PRINT AT 0,0;YMAX
  80 FOR g=1 TO 12
  90 PRINT AT 21,2*g;g
 100 FOR n=1 TO V(g)
 105 INK 2
 110 PRINT AT 20-n,2*g;"█"
 120 NEXT n
 122 FOR n=1 TO C(g)
 123 INK 1
 124 PRINT AT 20-n,2*g+1;"█"
 126 NEXT n
 130 NEXT g
 140 INK 7: PAPER 9
 150 STOP
 500 FOR I=1 TO 12
 510 LET V(I)=V(I)*19/YMAX
 520 LET C(I)=C(I)*19/YMAX
 530 NEXT I
 540 RETURN
```

REDEFINIÇÃO DE CARACTERES

SPECTRUM 16/48 K

DEFINIÇÃO DE CARACTERES (UDG)

In. ZX COMPUTING, Abril/Maio 84
Adapt. e Trad.: J. MAGALHÃES

Devido ao grande interesse em diferentes tipos de caracteres, apresentamos este programa que permite a alternativa da utilização dos caracteres em texto ou em títulos.

Além da redefinação de caracteres, pode definir gráficos UDG (User Defined Graphics) com diferentes utilizações.

Em qualquer das alternativas, a entrada de dados terá de ser em código decimal, tal como são apresentados no écran. A Tabela de Conversão que publicamos irá, concerteza, facilitar-lhe o trabalho.

Como verificará, os caracteres são apresentados com um tamanho 8 vezes superior ao normal, para melhor acompanhar o desenvolvimento de cada caracter ou gráfico.

O PROGRAMA

É usada uma pequena rotina em código máquina para copiar os caracteres da ROM e colocá-los na memória RAM. Note que o programa faz as devidas alterações da RAMTOP através do comando CLEAR.

Para evitar acidentes, o programa permite-lhe fazer BREAK e de novo RUN, sem perder os caracteres que redefiniu. No entanto, tome atenção à opção 1 do MENU principal que, se utilizada, lhe apagará todos os caracteres — todo o seu trabalho está perdido!

Para melhor compreensão do programa, apresentam-se as diferentes secções em que está dividido.

(Isto não foi introduzido no programa em linhas "REM staments" devido aos utilizadores do Spectrum 16K).

390 — 690 — MENU
710 — 740 — Desenho do quadrado (8/8)
810 — 860 — Aumento do tamanho normal dos caracteres (8 vezes)
1010 — 1090 — Apresentação dos caracteres na ROM
1210 — 1230 — Apresentação dos caracteres na RAM
1420 — 1550 — Alteração dos caracteres
1620 — 1700 — Apagar um caracter
1810 — 1880 — Apresentação dos gráficos definidos (UDG)
2010 — 2090 — Alteração dos gráficos (UDG)
2210 — 2330 — Visão completa dos caracteres e dos gráficos
2410 — 8999 — Rotinas de gravação (SAVE) e carregamento (LOAD)
9010 — 9999 — Colocação dos gráficos (UDG)

NOTA: Em código binário, o "0" (zero) corresponde ao quadrado branco e "1" ao quadro preenchido a negro.
P.ex.: ao código binário 01010010, corresponde o código decimal (DEC) "82".

```
        ALTERAR CARACTERES

              MENU

        CRIAR CARACTERES

1 Apaga todos os caracteres
2 Examinar caracter na ROM
3 Exam. caracter na memoria RAM
4 Alterar um caracter
5 Apagar apenas um caracter.

      DEFINIR GRAFICOS (UDG)

6 Ver caracter UDG
7 Alterar caracter UDG

8 Ver todos os caracteres
9 Rotinas LOAD/SAVE
```

```
  10>LET q=PEEK 23675+256*PEEK 2
3676: LET q=q-801
  20 POKE 23609,20
  40 CLEAR q
  50 LET q=PEEK 23730+256*PEEK 2
3731: LET q=q+31
  60 GO SUB 9000: GO SUB 9200
  70 GO SUB 140: IF PEEK 23681<>
1 THEN LET i$="1": GO TO 520
  80 GO TO 400
  90 STOP
 100 POKE 23606,0: POKE 23607,60
 110 RETURN
 140 LET r=q/256: POKE 23606,(r-
INT r)*256: POKE 23607,INT r-1
 150 RETURN
 400 BORDER 7: PAPER 7: INK 2: C
LS : GO SUB 100
 410 PRINT BRIGHT 1; INVERSE 1;
INK 1;TAB 7;"ALTERAR CARACTERES"
;TAB 31;" ";TAB 31;" "
 420 PRINT 'TAB 14; INK 1;"MENU"
 430 PRINT 'TAB 8; INK 3;"CRIAR
CARACTERES"
 440 PRINT '"1 Apaga todos os ca
racteres"'"2 Examinar caracter n
a ROM"'"3 Exam. caracter na memo
ria RAM"
 450 PRINT "4 Alterar um caracte
r"'"5 Apagar apenas um caracter.
"
 460 PRINT ' INK 3;TAB 6;"DEFINI
R GRAFICOS (UDG)"
 470 PRINT '"6 Ver caracter UDG"
'"7 Alterar caracter UDG"
 480 PRINT INK 3;'"8 Ver todos o
s caracteres"'"9 Rotinas LOAD/SA
VE"
```

```
 490 INK 0: PRINT #0; BRIGHT 1;
INK 7; PAPER 4;TAB 5;"A tua opca
o...";TAB 31;" "
 500 LET i$=INKEY$: IF i$<CHR$ 4
9 OR i$>CHR$ 57 THEN GO TO 500
 520 BEEP .1,CODE i$-64
 530 IF i$="1" THEN LET a=USR (q
-30): POKE 23681,1: GO TO 400
 540 IF i$="2" OR i$>"5" THEN GO
 SUB 100: GO TO 560
 550 GO SUB 140
 560 CLS : LET q$=("apresentacao
 do.caracter" AND i$<"6")+("Cara
cteres UDG" AND (i$="6" OR i$="7
"))+("Visao completa" AND i$="8"
)+("LOAD/SAVE rotinas" AND i$="9
")
 570 PRINT INK 3;AT 0,15-(LEN q$
/2);q$''
 580 GO SUB ((VAL i$)*200+600)
 600 PRINT #0; INK 1;"ENTER regr
essar ao MENU"
 620 IF INKEY$="" THEN GO TO 620
 630 IF INKEY$=CHR$ 13 THEN BEEP
 .1,12: GO TO 400
 650 GO TO 520
 690 STOP
 710 PLOT 128,128
 720 INK 1: DRAW 64,0: DRAW 0,-6
4: DRAW -64,0: DRAW 0,64
 740 INK 0: RETURN
 810 LET z=w: LET y=128
 820 INK 1: PRINT TAB 16;
 830 LET x=INT (z/y): PRINT CHR$
 164;: IF x=1 THEN PRINT CHR$ 8;
"█";: LET z=z-y
 850 LET y=y/2: IF y<1 THEN INK
0: BRIGHT 0: RETURN
 860 GO TO 830
1010 INPUT "Qual caracter? ";a$:
 LET a=CODE a$: IF a<32 OR a>127
 OR LEN a$<>1 THEN GO TO 1010
1020 LET a=(a-32)*8+15616
1030 GO SUB 700
1040 FOR f=1 TO 16: PRINT a$;" "
;: NEXT f
1060 PRINT ''"Codigo DEC.:"
1070 FOR f=a TO a+7: LET w=PEEK
f: PRINT 'f;"=";w;: GO SUB 800:
NEXT f
1080 PRINT '': FOR f=1 TO 16: PR
INT a$;" ";: NEXT f
1090 RETURN
1210 INPUT "Qual caracter? ";a$:
 LET a=CODE a$: IF a<32 OR a>127
 OR LEN a$<>1 THEN GO TO 1210
1220 LET a=(a-32)*8+q
1230 GO TO 1030
1420 INPUT "Caracter a ser alter
ado..";a$
1430 LET a=CODE a$: IF a>127 OR
a<32 OR LEN a$>1 THEN GO TO 1420
1440 PLOT 128,152: GO SUB 720: L
ET c=(a-32)*8+q
1460 FOR f=c TO c+7: LET w=PEEK
f: PRINT 'w;: PRINT TAB 10;" " A
ND f/2=INT (f/2);a$;: GO SUB 800
: NEXT f
1480 PLOT 128,80: GO SUB 720
1500 PRINT INK 3;''"Codigo DEC.:"
;: FOR f=0 TO 7
1510 INK 2: INPUT w: IF w<0 OR w
>255 OR w<>INT w THEN GO TO 1510
1520 POKE c+f,w: PRINT 'w;TAB 10
;" " AND f/2=INT (f/2);a$;: GO S
UB 800
1530 NEXT f
1540 PRINT '': FOR f=1 TO 16: PR
INT CHR$ a;" ";: NEXT f
1550 RETURN
1620 INPUT "Caracter a ser apaga
do...";a$
1630 LET a=CODE a$: IF a>127 OR
a<32 OR LEN a$>1 THEN GO TO 1620
1640 PLOT 128,152: GO SUB 720: L
ET c=(a-32)*8+q: LET a=(a-32)*8+
15616
1650 FOR b=1 TO 2
1660 FOR f=0 TO 7: PRINT ';: LET
 w=PEEK (c+f): IF f=0 OR f=7 THE
N PRINT TAB 0;: FOR g=1 TO 5: PR
INT a$;" ";: NEXT g
1670 GO SUB 800: NEXT f
1680 IF b=1 THEN FOR g=0 TO 7: P
OKE (g+c),PEEK (g+a): NEXT g: PR
INT AT 7,2;"foi:";AT 10,0: PLOT
128,80: GO SUB 720
1690 NEXT b: PRINT AT 16,2;"Agor
a:"
1700 RETURN
1810 INPUT "Qual caracter (UDG)?
";a$: LET a=CODE a$: LET a=a+(79
 AND a>64 AND a<91)+(47 AND a>96
 AND a<123): IF a<144 OR a>164 O
R LEN a$<>1 THEN GO TO 1810
1820 GO SUB 700: FOR f=1 TO 16:
PRINT CHR$ a;" ";: NEXT f
1830 PRINT ''"Cod.DEC. UDG:"
1840 LET c=USR a$
1850 FOR f=0 TO 7: LET w=PEEK (f
+c): PRINT 'w;: GO SUB 800
1860 NEXT f: PRINT ''
1870 FOR f=1 TO 16: PRINT CHR$ a
;" ";: NEXT f
1880 RETURN
2010 INPUT "Caracter de definir
?";a$: LET a=CODE a$: LET a=a+(7
9 AND a>64 AND a<91)+(47 AND a>9
6 AND a<123): IF a<144 OR a>163
THEN GO TO 2010
2020 LET c=USR a$(1): PLOT 128,1
52: GO SUB 720: FOR f=0 TO 7
2030 LET w=PEEK (f+c): PRINT 'w;
TAB 10;" " AND f/2=INT (f/2);CHR
$ a;: GO SUB 800: NEXT f
2040 PRINT INK 2;''"Novo cod. UDG
:";
2050 PLOT 128,80: GO SUB 720
2060 FOR f=0 TO 7: INK 2: INPUT
w: IF w<0 OR w>255 OR w<>INT w T
HEN GO TO 2060
2070 POKE (f+c),w: PRINT 'w;TAB
10;" " AND f/2=INT (f/2);CHR$ a;
: GO SUB 800: NEXT f
2080 PRINT '': FOR f=1 TO 16: PR
INT CHR$ a;" ";: NEXT f
2090 RETURN
2210 PRINT "caracteres na ROM:";
2220 INK 0: FOR g=1 TO 2
2240 FOR b=1 TO 6: PRINT : FOR f
=0 TO 14: PRINT " " AND (b/2=INT
 (b/2) AND f=0);CHR$ ((b*15)+f+1
7);" ";: NEXT f: NEXT b
2250 PRINT TAB 10;: FOR f=122 TO
 127: PRINT CHR$ f;" ";: NEXT f
2260 IF g=1 THEN INK 2: PRINT ''
caracteres na RAM:";: GO SUB 140
2270 NEXT g
2280 GO SUB 100
2290 INK 3: PRINT ''"Graficos def
inidos UDG:": FOR f=144 TO 164
2300 PRINT CHR$ f;" ";CHR$ 23+CH
R$ 11+CHR$ 0 AND f=159;
2310 NEXT f
2330 RETURN
2410 PRINT INK 1;''"A..Gravar car
ac. pre-definidos"''"B..Gravar g
raf. pre-def. UDG"''"C..Gravar e
ste programa"
2430 PRINT INK 2;''"D..Carregar
carac. pre-def."''"E..Carregar g
raf. pre-def. UDG"''"F..Carregar
 qualquer programa"
2450 LET a$=INKEY$: LET a=CODE a
```

```
$: LET a=a-(32 AND a>96 AND a<10
3): IF a<65 OR a>70 THEN GO TO 2
450
2460 LET b=3+(1 AND a>67)+((a-65
)*2): PRINT AT b,0; INK 8; FLASH
 1; OVER 1;TAB 30
2470 GO SUB (3000+((a-65)*100))
2480 IF INKEY$="n" OR INKEY$="N"
 THEN GO TO 2480
2490 RETURN
3000 SAVE "Char. Set"CODE q,768
3020 PRINT AT 20,6;"Verificar ?(
S/N)": LET a$=INKEY$
3030 IF a$="n" THEN GO TO 3080
3040 IF a$<>"s" THEN GO TO 3020
3050 PRINT AT 20,2;"Inicia o gra
vador....";AT 19,0: IF a<>67 THE
N VERIFY ""CODE
3060 IF a=67 THEN VERIFY ""
3080 RETURN
3100 SAVE "UDG's"CODE (q+769),16
8
3120 GO TO 3020
3200 LET a$="s": SAVE "Programa"
 LINE 1
3220 GO TO 3020
3300 PRINT AT 18,1;"Coloca o cab
o no EAR/Inicia o  Gravador";
3310 LOAD ""CODE q
3330 RETURN
3400 PRINT AT 18,1;"Cabo no EAR/
inicia o gravador";
3410 LOAD ""CODE (q+769)
3430 RETURN
3500 PRINT AT 18,12; FLASH 1;"AD
EUS"
3510 PRINT #1;"Qualquer tecla/in
icia o gravador"
3520 LOAD "": RUN
8999 STOP
9010 RESTORE 9000: FOR f=0 TO 7:
 READ a: POKE USR "u"+f,a
9020 NEXT f: RETURN
9030 DATA 128,128,128,128,128,12
8,128,255
9200 RESTORE 9200: LET r=q/256
9210 FOR f=(q-30) TO (q-30)+11:
READ a: POKE f,a: NEXT f: RETURN

9220 DATA 17,(r-INT r)*256,INT r
,33,0,61,1,0,3,237,176,201
9990 STOP
9999 GO SUB 100: POKE 23609,0
```

(Tabela de Conversão na pág. 19)

REGISTO DE VENDAS

SPECTRUM

ALEXANDRE SOUSA/CLUBE Z 80

O nosso amigo Abílio Lemos possui uma pequena firma de distribuição de electrodomésticos. Dado que esta actividade exige um controlo real sobre a marcha das vendas, solicitou-nos que fizéssemos o projecto de um programa para o SPECTRUM, que cumprisse esta função.

A listagem que publicamos possui o dimensionamento prévio para 15 produtos (linha 20, variável P) e 20 clientes (linha 10, variável c$).

O dimensionamento pode ser alterado, actuando nas linhas 10, 20 e 30.

Variáveis	Tipo	Dimensionamento	
c$	string	20	com 20 caracteres
p	numérica	15	índice 1 quantidade acum. índice 2 valor acum.
c	numérica	20	clientes; 15 produtos; índice 1: quantidade p/ produtos e clientes índice 2: valor p/ produto e cliente

O cliente está classificado por número ou nome e o produto está classificado apenas por código numérico entre 1 e 15.

O contador do número de fichas é a variável nf.

O programa permite conhecer, em qualquer altura, a quantidade acumulada por produto e o valor respectivo.

O lançamento de novos valores por cliente/produto permite dar entrada do preço unitário ou do valor total do produto facturado.

Se pretender a listagem na impressora, bastará repetir a parte do programa que vai das linhas 3120 até 3167, com substituição de PRINT por LPRINT e 3340-3390 (também alterando PRINT por LPRINT) dentro de um ciclo FOR-NEXT para repetir os clientes.

Sugerimos que seja introduzida na máquina a primeira parte do código BASIC desde a linha 10 até à linha 1060. Depois de fazer RUN, ficará com a possibilidade de usar a "←" que está definida nas linhas 999. Esta seta, que corresponde à acção da tecla 5 (shift), entra na linha 1070 com a seguinte acção:

a) GRAPHICS para entrar
b) Tecla 5
c) GRAPHICS para sair

Quando acabar de ter a listagem na máquina, faça SAVE "LINE 1".

Depois de ter o seu ficheiro criado ou actualizado, use sempre o mesmo programa e limite-se a dar entrada do ficheiro. A gravação do ficheiro obriga a 3 vezes (start the tape)! Para evitar isso, introduza no programa as seguintes linhas:

— 4148 POKE 23736, 181
— 4152 POKE 23736, 181
— 4158 POKE 23736, 181

e a gravação das variáveis será automática.

```
 10 DIM c$(20,20)
 20 DIM p(15,2)
 30 DIM c(20,15,2)
 90 LET nf=1
100 REM menu
105 POKE 23658,8
110 GO SUB 900
120 CLS : PRINT AT 1,10;"M  E  N  U": PRINT AT 3,10;"__________"
130 PRINT AT 6,0;"1 - Criar Fich. de Clientes"
140 PRINT AT 9,0;"2 - Actual.Valores/Quant."
150 PRINT AT 12,0;"3 - Listagem de Ficheiros"
155 PRINT AT 15,0;"4 - Grav./Entr. (Ficheiros)"
160 PRINT AT 20,0;"Optou p/ numero... ?"
170 PAUSE 0
180 IF INKEY$<"1" OR INKEY$>"4" THEN  GO TO 170
190 GO TO (VAL (INKEY$)*1000)
600 REM  mensagem
610 PRINT AT 21,0;"                              "
620 PRINT AT 21,0;m$
630 PAUSE 0
640 LET i$=INKEY$
650 LET cg=CODE (i$)
660 IF cg=8 OR cg=13 THEN  RETURN
670 GO TO 630
700 CLS : PRINT "cliente... ?": PRINT "__________"
710 INPUT "NUMERO...?";Y$
720 IF Y$="" THEN  GO TO 800
730 IF VAL (y$)>nf THEN  GO TO 700
```

```
740 LET n=VAL (y$): RETURN
800 INPUT "NOME...?";Y$
810 LET L=LEN (Y$)
820 FOR I=L TO 19
830 LET Y$=Y$+" "
840 NEXT I
850 FOR n=1 TO NF-1
860 IF C$(N)=Y$ THEN  RETURN
870 NEXT n
880 LET M$="INEXISTENTE....Enter p/repet."
885 GO SUB 600
890 GO TO 700
900 RESTORE
910 READ q$
920 FOR j=0 TO 7
930 READ r
940 POKE USR q$+j,r
950 NEXT j
960 RETURN
999 DATA "s",BIN 00011000,BIN 00110000,
    BIN 01100000,BIN 11111111,BIN 11111111,B
    IN 01100000,BIN 00110000,BIN 00011000
1000 REM criar
1010 CLS
1020 PRINT "Ficha num.  -  ";nf
1030 PRINT "____________________"

1040 PRINT AT1040>PRINT AT 6,0:"CLIENTE/NOME"
1050 INPUT Y$
1060 PRINT Y$
1070 LET M$="ENTER...OK !   "+"←"+" anular"
1080 GO SUB 600
1090 IF cq=13 THEN  GO TO 1110
1100 GO TO 1000
1110 LET c$(nf)=y$
1120 LET nf=nf+1
1130 LET M$="ENTER/cont. "+" "+" Regr/MENU"
1140 GO SUB 600
1150 IF cq=13 THEN  GO TO 1000
1160 GO TO 100
2000 REM LANC.
2010 GO SUB 700
2020 CLS
2030 PRINT C$(N): PRINT "________________________________"
2040 PRINT AT 10,0,"CODIGO/PROD.= ";
2050 INPUT J: IF J>15 THEN  GO TO 2050
2060 PRINT J
2070 PRINT "QUANTID. =";
2080 INPUT Q: PRINT Q
2090 PRINT "VALOR    =";
2100 INPUT V: PRINT V
2110 PRINT "Unitario/Total...?";
2112 PAUSE 0: LET Y$=INKEY$
2114 PRINT Y$
2115 IF Y$="T" THEN  GO TO 2130
2120 IF Y$="U" THEN  LET v=q*v: GO TO 2130
2125 GO TO 2112
2130 LET M$="ENTER...OK !   "+"←"+" anular"
2140 GO SUB 600
2150 IF cq=8 THEN  GO TO 2000
2160 LET c(n,j,1)=c(n,j,1)+q: LET P(J,1)=P(J,1)+Q
2170 LET c(n,j,2)=c(n,j,2)+v: LET P(J,2)=P(J,2)+V
2180 LET M$="ENTER/cont. "+" "+" Regr/MENU"
2190 GO SUB 600
2200 IF cq=8 THEN  GO TO 100
2210 LET M$="ENTER Cl.=  ("+" "+" Cl dif.)"
2220 GO SUB 600
2230 IF cq=8 THEN  GO TO 2000
2240 GO TO 2020
3000 REM listas
3010 CLS
3020 PRINT AT 10,0;"1 - PRODUTOS"
3030 PRINT AT 14,0;"2 - CLIENTES"
3040 PRINT AT 20,0,"Optou p/ numero... ?"
3050 PAUSE 0
3060 IF INKEY$<"1" OR INKEY$>"2" THEN  GO TO 3050
3070 GO TO (3000+VAL (INKEY$)*100)
3100 REM PRODUTOS
3105 LET ttp=0
3110 CLS
3120 PRINT "COD/PROD. -QUANTID.  - VALOR -"
3130 PRINT "______________________________"
3140 FOR J=1 TO 15
3150 PRINT J;TAB 10;P(J,1);TAB 22;P(J,2)
3155 LET ttp=ttp+p(j,2)
3160 NEXT J
3165 PRINT AT 17,0;"______________________________"
3167 PRINT TAB (20);ttp
3170 LET M$="ENTER p/regr. ao menu"
3180 GO SUB 600
3190>IF cg=13THEN GO TO 100
3195 GO TO 3180
3200 REM clientes
3210 CLS
3220 PRINT "LISTAGEM DE CLIENTES": PRINT "______________________"
3230 PRINT AT 10,0;"1 - Individual"
3240 PRINT AT 14,0;"2 - Colectiva"
3245 PRINT AT 20,0;"Optou p/ numero... ?"
3250 PAUSE 0
3260 IF INKEY$<"1" OR INKEY$>"2" THEN  GO TO 3250
3270 GO TO (3200+VAL (INKEY$)*100)
3300 REM
3310 GO SUB 700
3320 CLS
3330 PRINT "CLIENTE = ";C$(N)
3340 PRINT "______________________________"
3350 PRINT "COD/PROD.   QUANT.     VALOR"
3360 PRINT "______________________________"
3365 LET ttc=0
3370 FOR J=1 TO 15
3380 PRINT J;TAB 10;C(N,J,1);TAB 22;C(N,J,2)
3385 LET ttc=ttc+c(n,j,2)
3388 NEXT J
3390 PRINT TAB 20;"___________"  PRINT TAB 20;ttc
3392 LET M$="proximo...ENTER  "+"←"+"r/Menu"
3394 GO SUB 600
3396 IF cg=13 THEN  LET n=n+1: GO TO 3320
3398 GO TO 100
3400 REM
3410 CLS
3420 LET n=1
3430 GO TO 3320
4000 REM gravar
4100 CLS
4110 PRINT AT 10,0;"1 - ENTRADA DO FICHEIRO"
4112 PRINT AT 14,0;"2 - GRAVAR O FICHEIRO"
4114 PAUSE 0: IF INKEY$="1" THEN  GO TO 4200
4116 IF INKEY$="2" THEN  GO TO 4120
4118 GO TO 4114
4120 CLS : PRINT "GRAVAR OS FICHEIROS"
4130 PRINT AT 10,0,"PREPARAR O GRAVADOR"
4140 LET M$="ENTER quando estiver pronto!"
4145 GO SUB 600
4150 SAVE "fichas" DATA c$()
4155 SAVE "fichas-c" DATA c()
4160 SAVE "fichas-p" DATA p()
4170 GO TO 100
4200 REM LOAD
4205 CLS
4220 PRINT AT 10,0,"PREPARAR O GRAVADOR"
4230 LET M$="ENTER quando estiver pronto!"
4240 GO SUB 600
4250 LOAD "" DATA C$()
4260 LOAD "" DATA C()
4270 LOAD "" DATA P()
4280 GO TO 100
```

CALENDÁRIO

NEW BRAIN

Autor: PAULO CASTELO
PORTO

```
10  REM  ----------------------------------------------------------
20  REM  !  "*21:Calendario"           NB8405081          !
30  REM  !  28/3/1984    VERSAO 1ii    Paulo Castelo      !
40  REM  ----------------------------------------------------------
50  REM
60  REM  Computador NEWBRAIN AD  - Impressora SEIKOSHA GP-250X
70  REM  Linguagem  COMPILADOR INTERACTIVO  ANSI BASIC-80 PLUS
80  REM===========================================================
90  REM       Este programa imprime Calendarios
100 REM   deste seculo numa impressora qualquer
110 REM   com 80 colunas .
120 REM===========================================================
130 REM    LISTA DE CONSTANTES :
140 REM    Nome  ! Tipo  ! Conteudo
150 REM    ------+-------+--------------------
160 REM    tp$   ! ALFA  ! Tem o cabecalho de cada trimestre.
170 REM    me$() ! ALFA  ! Tem os nomes de cada mes.
180 REM    d()   ! REAL  ! Tem o numero de dias de cada mes.
```

```
190 REM    ln    | REAL | Tem o numero de linhas de cada folha.
200 REM    sp$   | ALFA | Tem 28 espacos para fins de tabulacao.
210 REM==========================================
220 REM    Funcao M7 : Da' o resto da divisao por sete
230 REM==========================================
240 REM    LISTA DE VARIAVEIS :
250 REM    Nome | Tipo | Conteudo
260 REM    ----+------+----------------
270 REM    z$    | ALFA | Tem o valor do ano e e' tambem acumulador.
280 REM    an    | REAL | Tem o numero do Ano do calendario.
290 REM    tm    | REAL | Tem o numero do Trimestre.
300 REM    ms    | REAL | Tem o numero do Mes.
310 REM    sm    | REAL | Tem o numero da semana do Mes.
320 REM    sm$   | ALFA | Tem todos os numeros dos dias de uma semana.
330 REM    i     | REAL | CONTADOR
340 REM    a$    | ALFA | Acumula a resposta `a Tecla de STOP.
350 REM    a     | REAL | Traduz numericamente a resposta a STOP.( 0<codigo<31 )
360 REM    di    | REAL | Tem o dia da semana quando tirada a funcao M7
370 REM    d(2)  | REAL | Tem o numero dos dias de Fevereiro e por isso...
380 REM          |      | ...ao contrario  dos outros  elementos de d()...
390 REM          |      | ...esta' constantemente a variar.
400 REM    dm    | REAL | Dia do Mes em que comeca a semana escolhida.
410 REM    sa    | REAL | Dia do Mes em que acaba a semana escolhida.
420 REM==========================================

995  REM ____________________________________________________[ Inicializacao ]
999  REM
1000 tp$="   DO SE TE QU QI SE SA   "  :  tp$=tp$+tp$+tp$
1010 DIM me$(12), d(12)
1020 ON BREAK GOTO 9900 : ln=72          : REM   linhas de cada folha ( ln >62 )
1030 CLOSE #8 : OPEN #8, 9, "2400"       : REM   ou OPEN #8, 8, "4800" p.exemplo
1040 sp$="                            "   : REM   28 Espacos
1050 DEF FN M7 (a) = 7*(a/7-INT(a/7))    : REM   'Argumento' MOD 7
1110 me$( 1)="JANEIRO"   : d(1)= 31
1120 me$( 2)="FEVEREIRO" : d(2)= 28
1130 me$( 3)="MARCO"     : d(3)= 31
1140 me$( 4)="ABRIL"     : d(4)= 30
1150 me$( 5)="MAIO"      : d(5)= 31
1160 me$( 6)="JUNHO"     : d(6)= 30
1170 me$( 7)="JULHO"     : d(7)= 31
1180 me$( 8)="AGOSTO"    : d(8)= 31
1190 me$( 9)="SETEMBRO"  : d(9)= 30
1200 me$(10)="OUTUBRO"   : d(10)=31
1210 me$(11)="NOVEMBRO"  : d(11)=30
1220 me$(12)="DEZEMBRO"  : d(12)=31

1995 REM ___________________________________________[ Impressao do Calendario ]
1999 REM
2000 PRINT " "
2010 LINPUT ("Entre o Ano do Calendario : ") z$
2020 IF NOT NUM(z$) PRINT "Deve ser um numero entre 0 e 99." : GOTO 2000
2030 an=VAL(z$) : z$=STR$(an+1900[6])
2040 PRINT #8, z$ : PRINT #8 : PRINT #8
2050 REM   Impressao de cada Trimestre
2060 FOR tm=1 TO 4
2070    FOR ms=3*tm-2 TO 3*tm
2080       PRINT #8,"    "; me$(ms); LEFT$(sp$,23-LEN(me$(ms)));
2090    NEXT ms : PRINT #8 : PRINT #8 : PRINT #8 : PRINT #8, tp$ : PRINT #8
2100    FOR sm=1 TO 6
2110       FOR ms=3*tm-2 TO 3*tm
2120          GOSUB 10000 : PRINT #8,"  "; sm$,"    ";
2130       NEXT ms : PRINT #8
2140    NEXT sm : PRINT #8 : PRINT #8 : PRINT #8 : PRINT #8
2150 NEXT tm
2160 FOR i=1 TO ln-63 : PRINT #8 : NEXT i : GOTO 2010

9895 REM ______________________________________________________________[ FIM ]
9899 REM
9900 PRINT " "
```

```
9910 PRINT "Carregou na Tecla STOP."
9920 PRINT " "
9930 LINPUT ("Quer continuar.C  ou acabar.A  ? ") a$
9940 a=ASC(a$) AND 31 : IF a$="" REPORT
9950 IF a=1 RESUME 9970
9960 RESUME
9970 ON BREAK GOTO 0 : END
9980 REM   GOTO 9990 serve para continuar se nao se responder nada `a linha 9930
9990 ON BREAK GOTO 9900 : CONTINUE

9995  REM _____________________[ sm$ = Semana n.'sm' do Mes n.'ms' do Ano n.'an' ]
9999  REM
10000 di=FN M7(an+1+INT((an-1)/4))    : REM   Dia da semana em que comeca o Ano
10010 IF an/4=INT(an/4) THEN d(2)=29 : REM   Caso do Ano Bissexto
10020 FOR i=1 TO ms-1
10030    di=di+d(i)
10040 NEXT i             : di=FN M7(di)    : REM   Dia da semana em que comeca o Mes
10050 sm$="" : dm=sm*7-6-di : sa=dm+6 : IF sa>d(ms) THEN sa=d(ms)
10060 IF dm<1 THEN sm$=LEFT$(sp$,3*(1-dm)) : dm=1
10070 FOR i=dm TO sa+.1      : REM   'sa+.1' para o ciclo terminar sempre em 'sa'
10080     sm$=sm$+LEFT$(STR$(i[2]),3)
10090 NEXT i : sm$=sm$+LEFT$(sp$,21-LEN(sm$)) : d(2)=28 :  RETURN
```

O CLUBE Z 80 apresenta as melhores felicitações ao Paulo Castelo pela magnífica forma de documentar o programa — linhas 10 a 995.

CONVERSÃO DE PROGRAMAS DO ZX81 → ZX SPECTRUM

Autor: FERNANDO PRECES
Sacavém

PARTE I

1.1 Introdução

Atendendo às centenas de anúncios para venda de ZX81, publicados em vários jornais, e aos milhares de ZX SPECTRUMs vendidos até agora, é previsível que alguns dos leitores deste Boletim já tenham trocado de computador. Também não é difícil prever que outros tantos aguardam uma oportunidade para o fazer.

Este texto tem como objectivo auxiliar o leitor nessa transição, visto que a vantagem adquirida na diferença de categoria entre as duas máquinas é perdida, tendo em conta o trabalho desperdiçado a nível de programação, pois o SPECTRUM rejeita a entrada de programas gravados pelo ZX81.

Os leitores que ainda não se desfizeram dos seus programas vão ter a oportunidade de os introduzir no SPECTRUM, com a ajuda de um monitor apropriado, um tanto moroso a programar mas de grande eficiência.

Após a conversão da simbologia, muitos desses programas precisarão de uns **retoques** para funcionarem em pleno; mas, ainda assim, haverá uma grande economia de tempo se os passar directamente das fitas aonde estão gravados para o interior do SPECTRUM.

1.2 — ANÁLISE DOS SÍMBOLOS

Como possivelmente o leitor que vendeu o seu ZX81 o fez conjuntamente com o respectivo Manual, será melhor analisarmos, um a um, os símbolos e gráficos das duas máquinas e as suas equivalências.

Começarei por vos dar um quadro geral dessa equivalência (fig. 1.2) e uma indicação dos símbolos que vão ser marginalizados ou substituídos no SPECTRUM.

SIMBOLOS	CODE ZX81	CODE SPECTR
□	0	32
▘	1	130
▝	2	129
▀	3	131
▖	4	136
▌	5	138
▞	6	137
▛	7	139
▒	8	-
░	9	-
░	10	-
"	11	34
£	12	96
$	13	36

:	14	58
?	15	63
(	16	40
)	17	41
>	18	62
<	19	60
=	20	61
+	21	43
-	22	45
*	23	42
/	24	47
;	25	59
,	26	44
.	27	46
0	28	48
1	29	49
2	30	50
3	31	51
4	32	52
5	33	53
6	34	54
7	35	55
8	36	56
9	37	57
A	38	65
B	39	66
C	40	67
D	41	68
E	42	69
F	43	70
G	44	71
H	45	72
I	46	73
J	47	74
K	48	75
L	49	76
M	50	77
N	51	78
O	52	79
P	53	80
Q	54	81

R	55	82
S	56	83
T	57	84
U	58	85
V	59	86
W	60	87
X	61	88
Y	62	89
Z	63	90
RND	65	165
INKEY$	66	166
PI	67	167
cursor ↑	112	11
cursor !	113	10
cursor <=	114	8
cursor =>	115	9
GRAPHICS	116	15
EDIT	117	7
NEWLINE	118	13
RUBOUT	119	12
Numero	126	14
CURSOR	127	-
█	128	143
▟	129	141
▙	130	142
▄	131	140
▜	132	135
▐	133	133
▚	134	134
▛	135	139
▒	136	-
▀	137	-
▄	138	-
139/191	INVERSAO DE VIDEO	
""	192	2* CODE 34
AT	193	172
TAB	194	173
CODE	196	175
VAL	197	176
LEN	198	177
SIN	199	178

COS	200	179
TAN	201	180
ASN	202	181
ACS	203	182
ATN	204	183
LN	205	184
EXP	206	185
INT	207	186
SQR	208	187
SGN	209	188
ABS	210	189
PEEK	211	190
USR	212	192
STR$	213	193
CHR$	214	194
NOT	215	195
*	216	94
OR	217	197
AND	218	198
<=	219	199
>=	220	200
<>	221	201
THEN	222	203
TO	223	204
STEP	224	205
LPRINT	225	224
LLIST	226	225
STOP	227	226
SLOW	228	-
FAST	229	-
NEW	230	230
SCROLL	231	-
CONTINUE	232	232
DIM	233	233
REM	234	234
FOR	235	235
GO TO	236	236
GO SUB	237	237
INPUT	238	238
LOAD	239	239
LIST	240	240
LET	241	241
PAUSE	242	242
NEXT	243	243
POKE	244	244
PRINT	245	245
PLOT	246	246
RUN	247	247
SAVE	248	248
RANDOMIZE	249	249
IF	250	250
CLS	251	251
UNPLOT	253	-
CLEAR	253	253
RETURN	254	254
COPY	255	255

Verifica-se, pela figura apresentada, que uma grande parte da simbologia é compatível, diferindo somente nos códigos de referência o que, na conversão, apenas implica a substituição numérica desses códigos.
Vejamos os casos restantes:

a) Símbolos em **inverse-vídeo**

Para converter este tipo de símbolos, o SPECTRUM, como não recebe informação **de cor,** imprime-os a preto e branco, tal e qual os recebe.

b) Símbolos gráficos 8/10 e 136/138

Estes símbolos, programados no ZX81 para conseguir no écran o tom cinzento, também não tem siginificado no SPECTRUM que os transforma em gráficos a preto e branco.

c) Instruções SLOW e FAST

O SPECTRUM **trabalha** o BASIC a uma única velocidade (semelhante à velocidade FAST do ZX81), mas assegurando a todo o tempo a continuidade de imagem, pelo que estas duas instruções se vão ignorar.

d) Instrução SCROLL

O SCROLL no SPECTRUM é automático e o seu interpretador BASIC não reconhece essa instrução introduzida no teclado.

e) Instruções PLOT e UNPLOT

A instrução PLOT no SPECTRUM imprime um «pixel» que é 1/16 mais pequeno que o do ZX81. Isto quer dizer que as coordenadas gráficas são muito diferentes nos dois computadores.
O SPECTRUM não reconhece a instrução UNPLOT, visto que o **seu PLOT** permite imprimir ou apagar um «pixel» no écran.
As pequenas modificações a introduzir deverão ser feitas após a conversão, já com o programa memorizado no SPECTRUM, e serão abordadas ao longo deste artigo.

NOTA: Se algum leitor pretender a listagem do programa elaborado por FERNANDO PRECES para produzir o quadro de equivalências, pode solicitá-la ao CLUBE Z80.

(Cont. no próximo número)

TABELA DE CONVERSÃO

DEC.	BINÁRIO	HEX.	DEC.	BINÁRIO	HEX.	DEC.	BINÁRIO	HEX.	DEC.	BINÁRIO	HEX.
0	00000000	00	64	01000000	40	128	10000000	80	192	11000000	C0
1	00000001	01	65	01000001	41	129	10000001	81	193	11000001	C1
2	00000010	02	66	01000010	42	130	10000010	82	194	11000010	C2
3	00000011	03	67	01000011	43	131	10000011	83	195	11000011	C3
4	00000100	04	68	01000100	44	132	10000100	84	196	11000100	C4
5	00000101	05	69	01000101	45	133	10000101	85	197	11000101	C5
6	00000110	06	70	01000110	46	134	10000110	86	198	11000110	C6
7	00000111	07	71	01000111	47	135	10000111	87	199	11000111	C7
8	00001000	08	72	01001000	48	136	10001000	88	200	11001000	C8
9	00001001	09	73	01001001	49	137	10001001	89	201	11001001	C9
10	00001010	0A	74	01001010	4A	138	10001010	8A	202	11001010	CA
11	00001011	0B	75	01001011	4B	139	10001011	8B	203	11001011	CB
12	00001100	0C	76	01001100	4C	140	10001100	8C	204	11001100	CC
13	00001101	0D	77	01001101	4D	141	10001101	8D	205	11001101	CD
14	00001110	0E	78	01001110	4E	142	10001110	8E	206	11001110	CE
15	00001111	0F	79	01001111	4F	143	10001111	8F	207	11001111	CF
16	00010000	10	80	01010000	50	144	10010000	90	208	11010000	D0
17	00010001	11	81	01010001	51	145	10010001	91	209	11010001	D1
18	00010010	12	82	01010010	52	146	10010010	92	210	11010010	D2
19	00010011	13	83	01010011	53	147	10010011	93	211	11010011	D3
20	00010100	14	84	01010100	54	148	10010100	94	212	11010100	D4
21	00010101	15	85	01010101	55	149	10010101	95	213	11010101	D5
22	00010110	16	86	01010110	56	150	10010110	96	214	11010110	D6
23	00010111	17	87	01010111	57	151	10010111	97	215	11010111	D7
24	00011000	18	88	01011000	58	152	10011000	98	216	11011000	D8
25	00011001	19	89	01011001	59	153	10011001	99	217	11011001	D9
26	00011010	1A	90	01011010	5A	154	10011010	9A	218	11011010	DA
27	00011011	1B	91	01011011	5B	155	10011011	9B	219	11011011	DB
28	00011100	1C	92	01011100	5C	156	10011100	9C	220	11011100	DC
29	00011101	1D	93	01011101	5D	157	10011101	9D	221	11011101	DD
30	00011110	1E	94	01011110	5E	158	10011110	9E	222	11011110	DE
31	00011111	1F	95	01011111	5F	159	10011111	9F	223	11011111	DF
32	00100000	20	96	01100000	60	160	10100000	A0	224	11100000	E0
33	00100001	21	97	01100001	61	161	10100001	A1	225	11100001	E1
34	00100010	22	98	01100010	62	162	10100010	A2	226	11100010	E2
35	00100011	23	99	01100011	63	163	10100011	A3	227	11100011	E3
36	00100100	24	100	01100100	64	164	10100100	A4	228	11100100	E4
37	00100101	25	101	01100101	65	165	10100101	A5	229	11100101	E5
38	00100110	26	102	01100110	66	166	10100110	A6	230	11100110	E6
39	00100111	27	103	01100111	67	167	10100111	A7	231	11100111	E7
40	00101000	28	104	01101000	68	168	10101000	A8	232	11101000	E8
41	00101001	29	105	01101001	69	169	10101001	A9	233	11101001	E9
42	00101010	2A	106	01101010	6A	170	10101010	AA	234	11101010	EA
43	00101011	2B	107	01101011	6B	171	10101011	AB	235	11101011	EB
44	00101100	2C	108	01101100	6C	172	10101100	AC	236	11101100	EC
45	00101101	2D	109	01101101	6D	173	10101101	AD	237	11101101	ED
46	00101110	2E	110	01101110	6E	174	10101110	AE	238	11101110	EE
47	00101111	2F	111	01101111	6F	175	10101111	AF	239	11101111	EF
48	00110000	30	112	01110000	70	176	10110000	B0	240	11110000	F0
49	00110001	31	113	01110001	71	177	10110001	B1	241	11110001	F1
50	00110010	32	114	01110010	72	178	10110010	B2	242	11110010	F2
51	00110011	33	115	01110011	73	179	10110011	B3	243	11110011	F3
52	00110100	34	116	01110100	74	180	10110100	B4	244	11110100	F4
53	00110101	35	117	01110101	75	181	10110101	B5	245	11110101	F5
54	00110110	36	118	01110110	76	182	10110110	B6	246	11110110	F6
55	00110111	37	119	01110111	77	183	10110111	B7	247	11110111	F7
56	00111000	38	120	01111000	78	184	10111000	B8	248	11111000	F8
57	00111001	39	121	01111001	79	185	10111001	B9	249	11111001	F9
58	00111010	3A	122	01111010	7A	186	10111010	BA	250	11111010	FA
59	00111011	3B	123	01111011	7B	187	10111011	BB	251	11111011	FB
60	00111100	3C	124	01111100	7C	188	10111100	BC	252	11111100	FC
61	00111101	3D	125	01111101	7D	189	10111101	BD	253	11111101	FD
62	00111110	3E	126	01111110	7E	190	10111110	BE	254	11111110	FE
63	00111111	3F	127	01111111	7F	191	10111111	BF	255	11111111	FF

NOVOS PROGRAMAS — SPECTRUM

JOGOS

Código* / Preços

- BLUE THUNDER (48 K) — Tens de pilotar um helicóptero em 5 missões diferentes. As tuas munições e o combustível são limitados, assim como o n.º de passageiros — **I/400$00**
- BIRDS AND BEES (48 K) — Vais controlar a abelha «Boris» que pretende colher néctar para a sua colmeia. Quanto mais carregar, mais lento será o voo. Atenção aos seres intrusos! — **IP/400$00**
- CITY (48 K) — Terás de construir a tua própria cidade e a sua vida: casas, lojas, barcos, pubs, etc. Mas não é muito simples essa vida porque na cidade também é necessário pagar impostos — **IP/400$00**
- THE GUARDIAN (48 K) — Evita os ataques dos «trackers», «antiminer», «swirls» e «snarks» e obterás bónus se os conseguires vaporizar com o Laser ou o «star-smasher». — **IP/400$00**
- JET SET WILLY (48 K) — Tens de percorrer diferentes divisões da casa, apanhando o maior número possível de objectos. — **IP/400$00**
- MICROMOUSE (16 K) — A tua missão é ajudares Micromouse a proteger os seus programas dos «Bugs» que pretendem destruí-los. Combate-os usando «Datakill». — **IP/400$00**
- NIGHT GUNNER (48 K) — És o piloto de um avião que terá de passar por 30 missões. Possuis uma arma para te defenderes e destruíres as posições inimigas. 4 níveis de dificuldade. — **I/400$00**
- PARATROOPERS (48 K) — Impede o salto dos paraquedistas, disparando sobre os helicópteros. Se algum paraquedista cair sobre tua metralhadora ou se deixares que 5 deles atinjam o solo, o jogo termina. — **IP/400$00**
- PEDRO (48 K) — Vais apanhar e plantar sementes, substituindo aquelas que foram comidas pelas diferentes criaturas que irão aparecendo. — **IP/400$00**
- ROMAN EMPIRE (48 K) — Um jogo de estratégia baseado no governo de legiões da era do Império Romano. Encarnarás o comandante das legiões e tens às tuas ordens 5 generais. — **IP/400$00**
- SLOT MACHINE (48 K) — Simulação da já conhecida Slot Machine. — **I/400$00**
- SNOW MAN (48 K) — Vais passear pelo jardim, tentando apanhar objectos e principalmente neve para fazeres o teu boneco. Come para ficares forte e apanha os rebuçados para obteres pontos extra. — **IP/400$00**

UTILITÁRIOS E DIDÁCTICOS

- ALEMÃO (48 K) — Auxiliar do estudo dos plurais e conjugação de verbos da língua alemã. — **P/600$00**
- CDU (48 K) — É 1 editor de caracteres definíveis pelo utilizador, aliando a facilidade de edição (comandos de tecla única, definição de caracteres através de movimentos do cursor) e a possibilidade de edição simultânea de 6 caracteres visualizados no écran, quer em tamanho real, quer escalado. O lado 2 da cassete contém 1 exemplo de construção e utilização de CDU's. — **P/600$00**
- CIRCUITOS LÓGICOS (48 K) — Permite fazer o traçado de circuitos lógicos no écran e usar símbolos lógicos. — **P/500$00**
- CONTA BANCÁRIA (48 K) — Organizar o extracto da conta bancária com toda a informação sobre valor, número de cheque, pesquisa para nome do destinatário, etc. — **P/500$00**
- ESTRUTUTA ATÓMICA (48 K) — Distribuição dos electrões nas respectivas órbitas. Dados sobre a Tabela Periódica. — **P/500$00**
- ELECTRONICS (48 K) — Permite-lhe desenhar um circuito electrónico no écran e testar o seu funcionamento, mesmo despois de eliminar ou substituir partes. Díodos, Transistores, Tiristores, Circuitos Lógicos, etc. — **I/700$00**
- ESTRUTURAS RETICULADAS (48 K) — Determina reacções nos apoios, esforços normais nas barras; Cálculo de esforços pelo método dos nós; Resolução de sistemas de equação pelo método de Gauss; Determinação deslocamentos de nós pelo Teorema Castigliano. — **P/500$00**
- FRANCÊS (48 K) — Estudo de plurais e conjugação de verbos da língua francesa. — **P/600$00**
- INGLÊS I (48 K) — Auxiliar para estudo das conjugações de verbos ingleses. — **P/600$00**
- SALÁRIOS (48 K) — Permite processar salários de pessoal até 80 fichas. Calcula descontos. Executa mapas de valores líquidos e ilíquidos e emite recibos. Pode usar impressoras Sinclair/Timex/Seikosha/Epson. — **P/500$00**
- SEGURANÇA (48 K) — Permite a realização de cópias de segurança das cassetes, incluindo os blocos «headerless». Tem capacidade para copiar, garantidamente, programas, arrays, blocos de código binário de até 49 089 bytes. — **P/1 200$00**

* Códigos: **P** — programa e instruções em português
I — programa e instruções em inglês
IP — programa em inglês e instruções em português

NOVOS LIVROS

- ADVANCED GRAPHICS WITH THE SINCLAIR SPECTRUM (ANGELL, I.O. e JONES, B.J., Macmillan,, Londres, 1983)

  — Operações gráficas. Rotinas para gráficos a 2 dimensões. Representação matricial. Caracteres gráficos. Construção de diagramas. Geometria. Projecções. Perspectivas.

  Preço (fotocópias) — 470$00

- THE COMPLETE SINCLAIR DATA BASE (ADAMS, BEARDSMORE, GILBERT, Ed. Landry)

  — Pequenas descrições de alguns jogos para ZX81 e Spectrum. Respostas a dúvidas frequentes sobre Hardware e Software.

  Preço (fotocópias) — 350$00

DESCONTO DE 20 % PARA SÓCIOS DO CLUBE — VENDAS NA SEDE DO CLUBE OU À COBRANÇA